View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Elsevier - Publisher Connector

Journal of Computer and System Sciences 78 (2012) 142–150

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

An improved two-party identity-based authenticated key agreement protocol using pairings

Marko Hölbl [*], Tatjana Welzer, Boštjan Brumen

Faculty of Electrical Engineering and Computer Science, University of Maribor, Smetanova ulica 17, 2000 Maribor, Slovenia

ARTICLE INFO

Article history:
Received 29 September 2008
Received in revised form 5 January 2011
Accepted 18 January 2011
Available online 22 January 2011

Keywords:
Authentication
Key agreement
Identity-based
Bilinear pairing
Cryptography
Security

ABSTRACT

Two-party authenticated key agreement protocols using pairings have gained much attention in the cryptographic community. Several protocols of this type where proposed in the past of which many were found to be flawed. This resulted in attacks or the inability to conform to security attributes. In this paper, we propose an efficient identity-based authenticated key agreement protocol employing pairings which employs a variant of a signature scheme and conforms to security attributes. Additionally, existing competitive and the proposed protocol are compared regarding efficiency and security. The criteria for efficiency are defined in this paper, whereas the criteria for security are defined by the fulfilment of security attributes from literature.

© 2011 Elsevier Inc. All rights reserved.

1. Introduction

Key establishment is a process with help of which two or more entities establish a session key. The key can be later used to achieve a cryptographic goal, like confidentiality or integrity. A common division of key establishment protocols can be as follows: in *key transport protocols*, one entity generates the key and transfers it to the other entity whereas in *key agreement protocols* both entities contribute data to establish a session key. Further suppose two honest entities A and B establish a common session key, namely participate in a key agreement protocol. We say that a key agreement protocol provides *implicit key authentication* if entity A is assured that no other entity besides entity B can learn the value of a particular secret key. Hence, a protocol providing implicit key authentication for all participating entities is called an *authenticated key agreement protocol* [20].

Another property of key agreement protocols is *key confirmation*. A protocol is said to provide key confirmation if entity A is assured that the other entity B is in possession of the secret key. If both implicit key authentication and key conformation are provided for all participating entities, the protocol is said to provide *explicit key authentication*. Hence, a key agreement protocol which provides explicit key authentication is referred to as *authenticated key agreement protocol with key conformation* [20]. Further details regarding key agreement protocols can be found in [24]

In this paper we focus on the first group of protocols, namely authenticated key agreement protocols. The concept of key agreement was first proposed by Diffie and Hellman [13]. However, their protocol is not secure against man-in-the-middle attack. After that many key agreement protocols were proposed, but all of them need a public key infrastructure (PKI), which requires high computational and storage efforts.

[*] Corresponding author. Fax: +386 2 2207272.
E-mail address: marko.holbl@uni-mb.si (M. Hölbl).

0022-0000/$ – see front matter © 2011 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcss.2011.01.002

In 1984, Shamir introduced the concept of identity-based cryptosystem [30] in which a user's public key is an easily calculated function of her/his identity (e.g. email address, phone numbers, office locations, etc.), while a user's private key can be calculated by a trusted authority, referred to as Private Key Generator (PKG). In the same paper, Shamir provided the first identity-based key construction based on the RSA problem, and presented an identity-based signature scheme. The identity-based public key cryptosystem simplifies the process of key management and thus can be an alternative to certificate-based public key infrastructure (PKI).

Later, in 2000, Joux [19] firstly demonstrated the construction of a key agreement protocol employing pairings. The first formally proven identity-based encryption scheme employing pairings was published by Boneh and Franklin [4]. Since then many identity-based key agreement protocols employing pairings have been proposed. In this paper we focus on two-party identity-based authenticated key agreement protocol employing pairing operations. Many protocols of this type were proposed [34,31,23,8,37], analyzed and some broken [35,11,40,32,38].

In the paper we propose an improved identity-based authenticated key agreement protocol using pairings for the two-party settings. The novel protocol is based on the signature scheme by Hess [18]. Furthermore, the proposed protocol is discussed, analyzed and evaluated regarding security and efficiency, and compared to competitive protocols (in view of efficiency and security).

The rest of the paper is organized as follows: Section 2 gives preliminary concepts, i.e. bilinear maps, the associated computational problems, the security attributes and efficiency criteria. In Section 3 the proposed protocol together with the corresponding efficiency and security analysis is described. Additionally, the efficiency and security of the proposed protocol and competitive protocols is discusses in Section 4. A conclusion is drawn in Section 5.

2. Preliminaries

In this section, we briefly describe the preliminary concepts and properties needed later in the paper; i.e. bilinear maps, computational problems, efficiency criteria and security attributes. All the concepts form the basis for identity-based public key infrastructure employing pairings.

Traditional PKI (public key infrastructure) is expensive mainly because of the infrastructure needed to manage and authenticate public keys, and the difficulty in managing multiple communities. It is not to be believed that identity-based public key cryptography would replace the conventional PKIs, but rather offers an alternative solution.

In identity-based public key cryptography, one's public key is predetermined by information that uniquely identifies them, such as their email address. Originally the idea of this concept was applied to simplify certificate management in e-mail systems. When A sends an e-mail to B, she simply encrypts the message using the public key string of B's e-mail (e.g. bob@email.com). No public key certificate for B has to be obtained by A. When B receives the encrypted mail, she contacts the Private Key Generator (PKG), authenticates herself and thus can obtain the private key from the PKG, which enables her to decrypt the e-mail. In contrast to existing PKI, A is able to send encrypted mail to B even if B has not set-up her public key certificate yet. A special case of identity-based public key cryptography (PKC) can be implemented using pairings. Furthermore, the concept of identity-based cryptography and pairings can be used to construct authenticated key agreement protocols.

2.1. Bilinear maps

We describe in a more general format the basic definition and properties of the pairing. More details can be found in Joux [19] and Boneh and Franklin [4].

Let P denote a generator of $\mathbb{G}_1$, where $\mathbb{G}_1$ is an additive group of large order q and let $\mathbb{G}_2$ be a multiplicative group with $|\mathbb{G}_1| = |\mathbb{G}_2|$. A pairing is a map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ which has the following properties:

1. *Bilinearity*:
   Given $Q, W, Z \in \mathbb{G}_1$, we have $\hat{e}(Q, W + Z) = \hat{e}(Q, W) \cdot \hat{e}(Q, Z)$ and $\hat{e}(Q + W, Z) = \hat{e}(Q, Z) \cdot \hat{e}(W, Z)$.
   Therefore for any $q, b \in \mathbb{Z}_q$:
   $$\hat{e}(aQ, bW) = \hat{e}(Q, W)^{ab} = \hat{e}(abQ, W) = \hat{e}(Q, abW) = \hat{e}(bQ, W)^{a}.$$
2. *Non-degenerative*: $\hat{e}(P, P) \neq 1$, where 1 is the identity element of $\mathbb{G}_2$.
3. The map $\hat{e}$ is efficiently computable.

In practice $\mathbb{G}_1$ is a subgroup of the group of points on an elliptic curve over a finite field, e.g. $E(\mathbb{F}_p)$. Then $\mathbb{G}_2$ is a subgroup of a multiplicative group of a related finite field. Usually $\mathbb{G}_1$ has around 2^{160} elements and $\mathbb{G}_2$ is a subgroup of $E(\mathbb{F}_{p^r})$, where r is the embedding degree and p^r has about 1024 bits.

The map $\hat{e}$ can be derived by modifying the Weil pairing [14] (both inputs are of the same cyclic group) or the Tate pairing [25] (related inputs are in the left-hand side of the pairing map) on a elliptic curve over $\mathbb{F}_p$. The computational effort of the Tate pairing is less than of the Weil pairing. However, both need to be modified since the pairing may always output $1 \in \mathbb{G}_2$ on the right side of the pairing.

A more detailed explanations of topics regarding bilinear maps, the Weil and Tate pairings, the aspects implementation and selection of suitable curves can be found in [4,5,15,16,19,25,36].

2.2. Computational problems

The security of the identity-based authenticated key agreement protocols discussed in this paper is based on the computation problem described bellow [4,8]. With the group described in Section 2.1, there are the following problems in elliptic curve cryptography:

1. *Discrete Logarithm (DL) problem*
   Given $P, Q \in \mathbb{G}_1$, find an integer n such that $P = nQ$ whenever such integer exists.
2. *Computational Diffie–Hellman (CDH) problem*
   Given a tuple $(P, aP, bP) \in \mathbb{G}_1$ for $a, b \in \mathbb{Z}_q^*$, find the element abP.
3. *Decision Diffie–Hellman (DDH) problem*
   Given a quadruple $(P, aP, bP, cP) \in \mathbb{G}_1$ for $a, b, c \in \mathbb{Z}_q^*$, decide whether $c = ab \bmod q$ or not [3].
4. *Bilinear Diffie–Hellman (BDH) problem*
   Let P be a generator of $\mathbb{G}_1$. The BDH problem in $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e} \rangle$ is that given $(P, xP, yP, zP) \in \mathbb{G}_1$ for some x, y, z chosen at random from $\mathbb{Z}_q^*$, compute $\hat{e}(P, P)^{xyz} \in \mathbb{G}_2$.

2.3. Security attributes

There are a number of ways in which an attacker can attempt to break a key agreement protocol, and when constructing a key agreement protocol, the designer must consider what types of attack the protocol must resist. Such analysis has led to the development of various desirable security attributes for key agreement protocols. In order to get a sound key agreement protocol, we need to define properties, which are described in detail in [1,2,8,26]. Here we assume A and B are two honest entities. It is desired for authenticated key agreement protocols to possess the following security attributes:

- *Known-key security.* In each round of a key agreement protocol, a unique session key should be generated. Each key generated in one protocol round is independent and should not be exposed if other session keys are compromised, i.e. the compromise of one session key should not compromise other session keys.
- *Forward secrecy.* If the long-term private key of one or more entities are compromised, the secrecy of previously established session keys should not be affected. We say that a protocols features *partial forward secrecy* if some but not all of the entities' long-term keys can be corrupted without compromising previously established session keys, and we say that a protocols features *perfect forward secrecy* if the long-term keys of all the entities may be corrupted without compromising any session key previously established by these entities.
- *Key-compromise impersonation.* Suppose that the long-term secret key of one participating entity is disclosed (e.g. A). Obviously, an adversary who knows this secret key can impersonate the entity to other participating entities (e.g. A to B). However, it is desired that this disclosure does not allow the adversary to impersonate other entities (e.g. B) to the entity whose long-term secret key was disclosed (e.g. A).
- *Unknown key-share resilience.* After the protocol run, one entity (e.g. A) believes she shares a key with the other participating entity (e.g. B), while the other entity (e.g. B) mistakenly believes that the key is instead shared with an adversary. Therefore, a sound authenticated key agreement protocol should prevent the unknown key-share situation.
- *Key control.* The key should be determined jointly by both participating entities (e.g. A and B). None of the participating entities (e.g. A or B) can control the key alone.

2.4. Efficiency properties

Since an efficiency analysis and a comparison of the proposed protocols and competitive protocol are given, criteria used to evaluate efficiency must be defined. Additionally, the criteria are used to evaluate and compare efficiency of identity-based authenticated key agreement protocols.

There are two main groups of efficiency criteria: criteria for evaluating computational effort and criteria for evaluating communicational effort. The criteria for evaluating computational effort is defined as the number of computations conducted in every protocol run per participating entity. These computations include:

- pairing operations,
- scalar multiplications,
- exponentiations,
- additions, and
- hash operations.

Additionally, pre-computations of pairing operations in protocols are included. They can be computed in an off-line manner and thus reduce the computational effort. The computations can be divided into two classes dependent on the effort required.

- *Less expensive* operations include additions and hash operations.
- *More expensive* operations include pairing operations, scalar multiplications and exponentiations.

Note that according to [5], the effort to evaluate one pairing operation is approximately equal to the effort of computing three scalar multiplications.

When evaluating and comparing the protocols from the *communicational* point of view, *number of passes* per protocol participant and *number of rounds* per protocol participant are considered (used as criteria). When designing protocols we want to achieve *minimal number* of both.

3. Proposed two-party identity-based authenticated key agreement protocol using pairings

In 2003, Hess [18] proposed an identity-based signature scheme using pairing operations. We will show that his signature scheme can be employed to construct an identity-based authenticated key agreement protocol

The proposed improved two-party identity-based authenticated key agreement protocol similarly to other identity-based authenticated key agreement protocols, requires a private key generator (PKG) and consists of three phases: `system setup`, `private key extraction`, and `key agreement` phase.

`System setup`. In this phase the Private Key Generator (PKG) constructs two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ and a map $\hat{e}:\mathbb{G}_1\times\mathbb{G}_1\to\mathbb{G}_2$. Next it computes a cryptographic hash function $H:\mathbb{Z}_q^*\to\mathbb{G}_1$, a generator (primitive root) $P\in\mathbb{G}_1$, a random integer $s\in\mathbb{Z}_q^*$ as PKG's private key and PKG's public key as $P_{PKG}=sP$. All elements are of order q. Finally, parameters $\langle\mathbb{G}_1,\mathbb{G}_2,\hat{e},P,P_{PKG},H\rangle$ are published, whereas the PKG's secret key s is kept secret.

`Private key extraction`. For a user with identity ID_i the public key is derived as $Q_i=H(ID_i)$ and the private key as $S_i=sQ_i$. Both parameters are computed by the PKG and afterwards S_i is issued to the entity via a secure channel.

`Key agreement`. Since signatures are used to authenticate the participating entities, a message is generated that is later used to derive the session key. The message is signed by the sender (entity A) and both the message itself and the signature are sent to the other participating entity – the receiver (entity B). In the same manner entity B generates a messages sings it and send it to A. Then both can verify the signature and compute the session key.

1. A: $a\in\mathbb{Z}_q^*$,
   B: $b\in\mathbb{Z}_q^*$,
2. $A\to B$:$T_A=aP$, $r_A=\hat{e}(S_A,P)^a$, $V_A=H(T_A,r_A)$, $U_A=V_AS_A+aS_A$,
   $B\to A$: $T_B=bP$, $r_B=\hat{e}(S_B,P)^b$, $V_B=H(T_B,r_B)$, $U_B=V_BS_B+bS_B$,
3. A: $r'_B=\hat{e}(U_B,P)\cdot\hat{e}(Q_B,-P_{PKG})^{V_B}$,
   V_B =? $H(T_B,r'_B)$,
   $K_{AB}=aT_B=abP$,
4. B: $r'_A=\hat{e}(U_A,P)\cdot\hat{e}(Q_A,-P_{PKG})^{V_A}$,
   V_A =? $H(T_A,r'_A)$,
   $K_{BA}=bT_A=abP$,
5. A,B: $K=abP$.

Observe that the verification holds because $r'_A=\hat{e}(U_A,P)\cdot\hat{e}(Q_A,-P_{PKG})^{V_A}=\hat{e}(V_AS_A+aS_A,P)\cdot\hat{e}(Q_A,-sP)^{V_A}=\hat{e}(V_AS_A+aS_A,P)\cdot\hat{e}(S_A,-P)^{V_A}=\hat{e}(S_A,P)^a=r_A$.

3.0.1. Protocol properties

Analyzing the computational efficiency of the protocol the following can be observed:

- For message exchange, each participant has to compute 2 scalar multiplications, 1 multiplication, 1 pairing operation, 1 exponentiation, 1 hash operation and 1 addition.
- To compute the session key 2 pairing operations, 1 scalar multiplication, 1 exponentiation and 2 hash operations are required.

The signature operation can be optimized by pre-computing $r_A=\hat{e}(S_A,P)^a$ for the specific S_A. Additionally, one additional pairing operation can be eliminated in the key computation phase, if a large number of verifications are to be performed; i.e. pre-computing $\hat{e}(Q_A,-P_{PKG})$.

If summed up each participant has to compute 3 pairing operations (2 can be precomputed), 3 scalar multiplications, 1 multiplication, 1 addition, 2 exponentiations and 3 hash operations.

In the proposed protocol the ephemeral keys a and b determine the sessions key $K = abP$. The security of the protocol is also based on the discrete logarithm problem and the bilinear Diffie–Hellman problem which are described in Section 2.2. If an adversary tries to derive the session key from the transferred messages $\{T_A, U_A, V_A\}$ or $\{T_B, U_B, V_B\}$, she would have to compute i from $T_i = iP$ or $V_i = V_i S_i + iS_i$ which would be equivalent to solving the discrete logarithm problem. From message $U_i = H(T_i, r_i)$ she is unable to derive any valuable information since only the hash value is transfered. If a secure hash functions is used, an adversary is prevented from deriving the original data that has been hashed [24]. Since two honest entities A and B have to know either S_A or S_B to complete the verification steps successfully and no other entity can derive the session key, the protocol provides *implicit key authentication*.

Let us further discuss the fulfilment of the other security attributes.

- *Known key security.* Suppose that the adversary learned a session key $K_1 = a_1 b_1 P$ of a previous protocol run. In each protocol run unique session keys are computed, which depend upon the ephemeral private keys a and b. Therefore the knowledge of previous keys does not enable the adversary to derive other (future) session keys and does not give the adversary any information which she could use to derive other session keys.
- *Forward secrecy.* When discussing (perfect) forward secrecy of the propose protocol, we have to consider that the session key is not linked to the long-term private keys of the users. Past session keys can only be computed using the ephemeral private keys, since the key is computed as $K = abP$. Additionally, past transfered messages do not help the adversary in obtaining past session keys.

  Even if an adversary would obtain both long-term private keys S_A and S_B, she would not be able to acquire one of the ephemeral private keys, e.g. a. Observe that it is not possible to derive a from the transfered messages $T_A = aP$, $V_A = H(T_A, r_A)$, $U_A = V_A S_A + aS_A$ since this would be equal to solving the discrete logarithm problem. The same is valid for messages sent by B. Hence the proposed protocol provides *perfect forward secrecy*.
- *Key-compromise impersonation resilience.* Suppose that A's long-time private key S_A is disclosed to an adversary who wants to impersonate B to A. However, the adversary can only replay since she does not know S_B and hence cannot generate a valid signature for B which would pass the verification process (i.e. V_B =? $H(T_B, r'_B)$).
- *Unknown key-share resilience.* To implement such an attack on the proposed protocol, the adversary is required to learn the private key of some entity. In the protocol only the entity who knows both the ephemeral secret key a and the long-term private key S_A can generate valid U_A and V_A. Thus our protocol offers *unknown key-share resilience*.
- *Key control.* Since the session key is determined by both entities, a single entity can not influence the outcome of the session key and thus enforce its session key.

4. Comparison with competitive protocols

In this section we preform a comparison of efficiency and security of the reviewed protocols. The following protocols are included: Smart's protocols with and without escrow [34] , Yi's protocol [42], Chen–Kudla's protocols with and without escrow [8], Scott's protocol [29], Shim protocol [31], Ryu-Yoon-Yoo's protocol [28], McCullagh–Barreto's escrow and escrowless protocols [23], Xie's protocols with and without escrow [40], Boyd–Mao–Peterson's protocol [6], Yuan–Li' protocol [43], Choie–Jeong–Lee's protocol I and II [10], Wang's protocol [37], Li–Yuan–Li's protocol I and II [21], Oh–Yoon–Yoo's protocol [27], Lim–Lee–Lee's protocol [22], Wang–Cao–Cheng–Coo's protocol [39], Huang–Cao's protocol [17], Chow–Choo's protocols [9] and the proposed protocol.

4.1. Efficiency comparison

The efficiency criteria for comparing the protocols is combined from two parts: computational and communicational effort (see Tables 1 and 2). Firstly the comparison of the protocols regarding computational effort is made. The criteria to evaluate and compare the proposed protocol and competitive protocols are discussed in Section 2.4.

Observe that only operations which are expensive from the computational point of view are evaluated. Additionally, the pre-computations of pairings are taken into consideration as they can reduce the computational demands of a protocol.

The proposed protocol requires a higher number of pairing operations as a result of the signature scheme used. However, the number of pairing can be lowered due to pre-computations. However, only few scalar multiplications are required which consequentially lowers the computational demands. As to [5] 3 scalar multiplication are equally expensive than 1 pairing operation. Regardless that some competitive protocols are more efficient, they do not conform to all the security attributes and/or lack in security because of known attacks. It is desirable for identity-based authenticated key agreement protocol to be secure and efficient at the same time.

Next the comparison regarding communicational efficiency is made for which the criteria is also discussed in Section 2.4. Since all included protocols require 1 round, we do no list the data in Table 2.

When the communicational effort of the protocols is compared, we can observe that the majority of protocols require 1 pass with exception of Boyd–Mao–Peterson's, Choie–Jeong–Lee I, Chow–Choo without escrow and the proposed protocol. The proposed protocol requires the transfer of 3 messages and hence is three-pass. However, communicational demands have less impact on the efficiency of the protocols than computational demands as the most communication channels are broadband.

Table 1
Computation effort per user.

Protocol	PairOp	ScMul	Exp	Add	Hash
Smart with escrow[†]	2	2	0	0	1
Smart without escrow	2	2	0	0	1
Yi	1	2	0	2	0
Chen–Kudla with escrow[†]	1	3	0	1	1
Chen–Kudla without escrow	1	3	0	1	1
Scott[†]	1	0	2	1	0
Shim[*]	1	2	0	2	1
Ryu–Yoon–Yoo	1	2	0	0	1
McCullagh–Barreto without escrow[†]	1	1	1	0	0
McCullagh–Barreto with escrow[†]	1	2	2	0	0
Xie with escrow[**]	1	3	1	0	0
Xie without escrow[**]	1	3	1	0	0
Boyd–Mao–Peterson[†]	1	2	0	0	2
Yuan–Li	1	3	0	2	1
Choie–Jeong–Lee I[***]	2	4	0	0	1
Choie–Jeong–Lee II	2	4	0	0	2
Wang	1	3	0	3	3
Li–Yuan–Li I	1	2	2	0	0
Li–Yuan–Li II	1	2	2	1	0
Oh–Yoon–Yoo[†]	1	3	0	0	2
Lim–Lee–Lee	1	3	0	0	2
Wang–Cao–Cheng–Coo	1	1	1	0	1
Huang–Cao	2	3	3	2	1
Chow–Choo with escrow[†]	1	3	0	2	1
Chow–Choo without escrow	1	4	0	2	1
proposed protocol[‡]	1(3)	3	2	1	3

[*] Broken due to the Man-in-the-Middle attack by Sun and Hsieh [35].

[**] Broken due to Li–Yuan–Li's man-in-middle attack [21], Li–Yuan–Li's and Shim's key compromise impersonation attack [21,32,33].

[***] Broken due to the Shim's signature forgery attack [33].

[†] Does not fulfill all desirable security properties.

[‡] 1 pairing operation required by the protocol if the pre-computation are taken into consideration:

PairOp – pairing operations,
Mul – scalar multiplications,
Exp – exponentiation in $\mathbb{G}_2$,
Add – point additions in $\mathbb{G}_1$,
Hash – map-to-point hash operation.

Table 2
Communicational effort per user.

Protocol	No. of passes
Smart with escrow	1
Smart without escrow	1
Yi	1
Chen–Kudla with escrow	1
Chen–Kudla without escrow	2
Scott	1
Shim	1
Ryu–Yoon–Yoo	1
McCullagh–Barreto without escrow	1
McCullagh–Barreto with escrow	1
Xie with escrow	1
Xie without escrow	1
Boyd–Mao–Peterson	2
Yuan–Li	1
Choie–Jeong–Lee I	2
Choie–Jeong–Lee II	1
Wang	1
Li–Yuan–Li I	1
Li–Yuan–Li II	1
Oh–Yoon–Yoo	1
Lim–Lee–Lee	1
Wang–Cao–Cheng–Coo	1
Huang–Cao	1
Chow–Choo with escrow	1
Chow–Choo without escrow	2
Proposed protocol	3

Table 3
Security attributes fulfilment.

Protocol	KKS	PeFS	PaFS	KCI	UKS
Smart with escrow	+	−	−[a]	+	+
Smart without escrow	+	+	+	+	+
Yi	+	+	+	+	+
Chen–Kudla with escrow	+	−	+	+	+
Chen–Kudla without escrow	+	+	+	+	+
Scott	+	+	+	−[b]	+
Shim[*]	+	+	+	+	+
Ryu–Yoon–Yoo	+	+	+	−	+
McCullagh–Barreto with escrow	+	+	+	−[c]	+
McCullagh–Barreto without escrow	+	−	+	+	+
Xie with escrow[**]	+	+	+	+	+
Xie without escrow[**]	+	+	+	+	+
Boyd–Mao–Peterson	+	+	+	−	+
Yuan–Li	+	+	+	+	+
Choie–Jeong–Lee I[***]	+	+	+	+	+
Choie–Jeong–Lee II	+	+	+	+	+
Wang	+	+	+	+	+
Li–Yuan–Li I	+	+	+	+	+
Li–Yuan–Li II	+	+	+	+	+
Oh–Yoon–Yoo	+	+	+	−[d]	+
Lim–Lee–Lee	+	+	+	+	+
Wang–Cao–Cheng–Coo	+	+	+	+	+
Huang–Cao	+	+	+	+	+
Chow–Choo with escrow	+	−	+	+	+
Chow–Choo without escrow	+	+	+	+	+
proposed protocol	+	+	+	+	+

[*] Broken due to the Man-in-the-Middle attack by Sun and Hsieh [35].
[**] Broken due to Li–Yuan–Li's man-in-middle attack [21], Li–Yuan–Li's and Shim's key compromise impersonation attack [21,32,33].
[***] Broken due to the Shim's signature forgery attack [33].
[a] Due to the attack by Shim [32].
[b] Due to the attack by Shim [7].
[c] Due to the key-compromise impersonation attack by Xie [40].
[d] Due to the attack by Lim, Lee and Lim [22].

KKS – Known-key secrecy,
PeFS – Perfect forward secrecy,
PaFS – Partial forward secrecy,
KCI – Key-compromise impersonation,
UKS – Unknown key-share,
KC – Key control.

4.2. Security comparison

Security attributes are the most important ones when dealing with key agreement protocols. We sum up the security attributes in Table 3. Details about the security attributes are described in Section 2.3. We do not list the property of key control, as all the protocols discussed in this paper conform to this property. Later we give published attacks or weaknesses for each protocol. Nevertheless if a protocols fulfils particular security attributes it can not be used, if a weakness or attack was published. Often a known attack leads to the fact that a protocol does not fulfil a specific security attribute.

Most protocols conform to the above security attributes (see Table 3). Protocols by Shim, Xie and Choie–Jeong–Lee protocol I have been broken due to attacks. Protocol which do not fulfil all the security attributes include: Smart's protocol [34], Chen–Kudla with escrow [8], Scott's protocol [29], McCullagh–Barreto's protocol [23], Boyd–Mao–Peterson's protocol [6], Oh–Yoon–Yoo's protocol [27] and Chow–Choo protocol with escrow [9]. It can be observed that the proposed protocol features all the security attributes.

4.3. Known attacks

Some of the reviewed protocols have been shown to have serious security flaws, which were exploited for attacks (see Table 4). For some of the presented protocols no attacks have been published so far: Boyd–Mao–Peterson's protocols [6], Yuan–Li's protocol [43], Choie–Jeong–Lee's protocol II [10], Wang protocol's [37] and Li–Yuan–Li's protocols [21]. An attack was presented on Shim's protocol [31] and McCullagh–Barreto's protocol [23]. Two attacks were published for Smart's [34] protocol. The worst case regarding published attacks applied to protocol of Ryu–Yoon–Yoo [28] and Xie's Protocol [41]. Some of the protocols were security patches of existing protocols and nevertheless they were shown to be flawed (i.e. Ryu–Yoon–Yoo's protocol [28]).

Table 4
Known attacks.

Protocol	Attacks
Smart with escrow	– Shim's attack (forward secrecy) [31] – Cheng et al.'s attack [12]
Smart without escrow	\
Yi	\
Chen–Kudla with escrow	– Cheng et al.'s attack [12]
Chen–Kudla without escrow	\
Scott	\
Shim	– Sun–Hsieh's man-in-the-middle attack [35]
Ryu–Yoon–Yoo	– Wang et al.'s key-compromise impersonation attack [37] – Yuan–Li's reveal attack [43] – Yuan–Li's key compromise impersonation attack [43]
McCullagh–Barreto with escrow	– Choo's attack [11]
McCullagh–Barreto without escrow	\
Xie – both protocols	– Li–Yuan–Li's man-in-middle attack [21] – Li–Yuan–Li's key compromise impersonation attack [21] – Shim's key compromise impersonation attack [32,33]
Boyd–Mao–Peterson	\
Yuan–Li	\
Choie–Jeong–Lee I	– Shim's signature forgery attack [33]
Choie–Jeong–Lee II	\
Wang	\
Li–Yuan–Li I	\
Li–Yuan–Li II	\
Oh–Yoon–Yoo	– Lim–Lee–Lee's Basic impersonation attack [22] – Lim–Lee–Lee's key compromise impersonation attack [22]
Lim–Lee–Lee	\
Wang–Cao–Cheng–Coo	\
Huang–Cao	\
Chow–Choo with escrow	\
Chow–Choo without escrow	\
Proposed protocol	\

When a protocol suffers from attacks, its use is questionable. Additionally, an attack causes the unattainability of specific security attributes.

5. Conclusion

In this paper we proposed an improved identity-based authenticated key agreement protocol employing pairing for the two-party setting. It employs signatures to authenticate the participating entities and verify the transfered messages. Moreover we discussed the efficiency and security of the proposed protocol and showed that it conforms to all desirable security attributes. Finally, we compared the proposed protocol and existing competitive protocols regarding efficiency and security and showed that the proposed protocol conforms to all security attributes and is at the same time efficient.

References

[1] M. Bellare, P. Rogaway, Entity authentication and key distribution, in: Advances in Cryptology – CRYPTO '93: 13th Annual International Cryptology Conference, in: Lecture Notes in Comput. Sci., vol. 773, Springer, New York, 1994.

[2] S. Blake-Wilson, D. Johnson, A. Menezes, Key agreement protocols and their security analysis (extended abstract), in: Proceedings of the 6th IMA International Conference on Cryptography and Coding, in: Lecture Notes in Comput. Sci., vol. 1355, Springer, New York, 1997, pp. 30–45.

[3] D. Boneh, The decision Diffie–Hellman problem, in: Proceedings of the Third Algorithmic Number Theory Symposium, in: Lecture Notes in Comput. Sci., vol. 1423, Springer, New York, 1998, pp. 48–63.

[4] D. Boneh, M. Franklin, Identity-based encryption from the Weil pairing, SIAM J. Comput. 32 (3) (2003) 586–615.

[5] P.S.L.M. Barreto, H.Y. Kim, B. Lynn, M. Scott, Efficient algorithms for pairing-based cryptosystems, in: Advances in Cryptology – Crypto 2002, Proceedings, in: Lecture Notes in Comput. Sci., vol. 2442, Springer, New York, 2002, pp. 354–368.

[6] C. Boyd, W.B. Mao, K.G. Paterson, Key agreement using statically keyed authenticators, in: Proceedings of Applied Cryptography and Network Security – ACNS, in: Lecture Notes in Comput. Sci., vol. 3089, Springer, New York, 2004, pp. 248–262.

[7] L. Chen, Z. Cheng, N.P. Smart, Identity-based key agreement protocols from pairings, Internat. J. Inform. Secur. 6 (4) (2007) 213–241.

[8] L. Chen, C. Kudla, Identity based authenticated key agreement protocols from pairings, in: Computer Security Foundations Workshop, IEEE, USA, 2003, pp. 219–233.

[9] Z. Cheng, L. Chen, On security proof of McCullagh Barretos key agreement protocol and its variants, Internat. J. Secur. Networks 2 (2007) 251–259.

[10] Y.J. Choie, E. Jeong, E. Lee, Efficient identity-based authenticated key agreement protocol from pairings, Appl. Math. Comput. 162 (1) (2005) 179–188.

[11] K.K.R. Choo, Revisit of McCullagh–Barreto two-party ID-based authenticated key agreement protocols, Internat. J. Netw. Secur. 1 (3) (2005) 154–160.

[12] Z. Cheng, M. Nistazakis, R. Comley, L. Vasiu, On The indistinguishability-based security model of key agreement protocols-simple cases, Cryptology ePrint Archive Report 2005/129, 2005.

[13] W. Diffie, M. Hellman, New directions in cryptography, IEEE Trans. Inform. Theory 22 (6) (1976) 644–654.

[14] T. Frey, M. Müller, H. Rück, The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems, IEEE Trans. Inform. Theory 45 (5) (1999) 1717–1719.

[15] S. Galbraith, Supersingular curves in cryptography, in: 7th International Conference on the Theory and Application of Cryptology and Information Security – CRYPTO'01, in: Lecture Notes in Comput. Sci., vol. 2248, Springer, New York, 2001, pp. 495–513.

[16] S.D. Galbraith, K. Harrison, D. Soldera, Implementing the Tate pairing, in: Proceedings of the 5th International Symposium on Algorithmic Number Theory – ANTS, in: Lecture Notes in Comput. Sci., vol. 2369, Springer, New York, 2002, pp. 324–337.

[17] H. Huang, Z. Cao, An ID-based authenticated key exchange protocol based on bilinear Diffie–Hellman problem, Cryptology ePrint Archive Report 2008/224, 2008.

[18] F. Hess, Efficient identity based signature schemes based on pairings, in: Selected Areas in Cryptography – SAC01, in: Lecture Notes in Comput. Sci., vol. 2595, Springer, New York, 2003, pp. 310–324.

[19] A. Joux, A one round protocol for tripartite Diffie–Hellman, in: 4th International Symposium on Algorithmic Number Theory, in: Lecture Notes in Comput. Sci., vol. 1838, Springer, New York, 2000, pp. 385–394.

[20] L. Law, A. Menezes, M.H. Qu, J. Solinas, S. Vanstone, An efficient protocol for authenticated key agreement, Des. Codes Cryptogr. 28 (2) (2003) 119–134.

[21] S. Li, Q. Yuan, J. Li, Towards security two-part authenticated key agreement protocols, Cryptology ePrint Archive Report 2005/300, 2005.

[22] M.H. Lim, S. Lee, H. Lee, Cryptanalytic flaws in Oh et al.'s ID-based authenticated key agreement protocol, Cryptology ePrint Archive Report 2007/415, 2007.

[23] N. McCullagh, P.S.L.M. Barreto, A new two-party identity-based authenticated key agreement, Cryptology ePrint Archive Report 2004/122, 2004.

[24] A. Menezes, P.C. Van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997.

[25] A. Menezes, T. Okamoto, S. Vanstone, Reducing elliptic curve logarithms to logarithms in a finite field, IEEE Trans. Inform. Theory 39 (5) (1993) 1639–1646.

[26] D. Nalla, K.C. Reddy, ID-based tripartite authenticated key agreement protocols from pairings, Cryptology ePrint Archive Report 2003/004, 2003.

[27] J.B. Oh, E.J. Yoon, K.Y. Yoo, An efficient ID-based authenticated key agreement protocol with pairings, in: Parallel and Distributed Processing and Applications, in: Lecture Notes in Comput. Sci., vol. 4742, Springer, 2007, pp. 1458–1463.

[28] E.K. Ryu, E.J. Yoon, K.Y. Yoo, An efficient ID-based authenticated key agreement protocol from pairings, in: Networking 2004, in: Lecture Notes in Comput. Sci., vol. 3042, Springer, New York, 2004, pp. 1458–1463.

[29] M. Scott, Authenticated ID-based key exchange and remote log-in with simple token and PIN number, Cryptology ePrint Archive Report 2002/164, 2002.

[30] A. Shamir, Identity-based cryptosystems and signature schemes, in: Advances in Cryptology – CRYPTO'84, Springer, New York, 1985, pp. 47–53.

[31] K. Shim, Efficient ID-based authenticated key agreement protocol based on Weil pairing, Electronics Lett. 39 (8) (2003) 653–654.

[32] K. Shim, Cryptanalysis of two ID-based authenticated key agreement protocols from pairings, Cryptology ePrint Archive Report 2005/357, 2005.

[33] S.H. Seo, K. Shim, Cryptanalysis of ID-based authenticated key agreement protocols from bilinear pairings (short paper), in: Information and Communications Security, in: Lecture Notes in Comput. Sci., vol. 4307, Springer, New York, 2006, pp. 410–419.

[34] N.P. Smart, Identity-based authenticated key agreement protocol based on Weil pairing, Electronics Lett. 38 (13) (2002) 630–632.

[35] H.M. Sun, B.T. Hsieh, Security analysis of Shim's authenticated key agreement protocols from pairings, Cryptology ePrint Archive Report 2003/113, 2003.

[36] E.R. Verheul, Evidence that XTR is more secure than supersingular elliptic curve cryptosystems, J. Cryptology 17 (4) (2004) 277–296.

[37] Y. Wang, Efficient identity-based and authenticated key agreement protocol, Cryptology ePrint Archive Report2005/108, 2005.

[38] S.B. Wang, Z.F. Cao, H.Y. Bao, Security of an efficient ID-based authenticated key agreement protocol from pairings, in: Parallel and Distributed Processing and Applications – ISPA2005, in: Lecture Notes in Comput. Sci., vol. 3759, Springer, New York, 2005, pp. 342–349.

[39] S. Wang, Z. Cao, Z. Cheng, K.K.R. Choo, Perfect forward secure identity-based authenticated key agreement protocol in the escrow mode, Cryptology ePrint Archive Report 2007/313, 2007.

[40] G. Xie, Cryptanalysis of Noel McCullagh and Paulo S.L.M. Barreto's two-party identity-based key agreement, Cryptology ePrint Archive Report 2004/308, 2004.

[41] G. Xie, An ID-based key agreement scheme from pairing, Cryptology ePrint Archive Report 2005/093, 2005.

[42] X. Yi, Efficient ID-based key agreement from Weil pairing, Electronics Lett. 39 (2) (2003) 206–208.

[43] Q. Yuan, S. Li, A new efficient ID-based authenticated key agreement protocol, Cryptology ePrint Archive Report 2005/309, 2005.